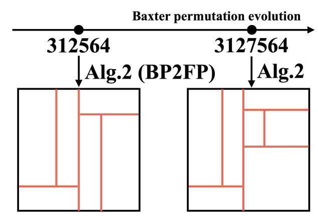
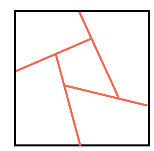
We thank reviewers for their thoughtful and positive feedback. **For significance**, we are encouraged that all reviewers (R1, R2, R3, R4) agree significance of our work, and also be glad that R3 recognizes insightful potential applications. **For novelty**, we are pleased that reviewers find that our paper is *creative* (R1), *finds a new angle* (R2) of model construction based on *a very different idea from the existing relevant methods* (R2), and *provides a novel tool (...) that not used (...) before* (R4). We will answer R3's concern about the difference between our model and the conventional rectangular tiling process (ICML2014) below. **For clarity**, we are glad that reviewers find that our paper is well written (R1,R3), *the idea is quite easy to follow* (R2), and *the new process makes sense* (R4). We will answer R2's main concern about how to obtain the evolution of the underlying floorplan partitioning according to our Baxter permutation process. **We address some specific comments below and will incorporate all the feedbacks in the revised paper.**

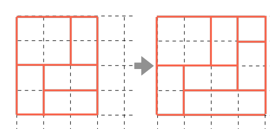
For R1 - We again appreciate your positive feedback. We agree with R1 in the sense that combinatorial stochastic processes involving permutations must be one of the promising directions of BNP for further studies and developments.



@R2 - **How to obtain the evolution of a floorplan partition (FP) according to the Baxter permutation (BP) process**: Instead of direct transformations from a FP with n blocks to a FP with $n+1$ blocks, the evolution of a FP is obtained only through the underlying evolution of a BP by using Algorithm 2 (mapping from BPs into FPs). For example, we consider an evolution of a BP from **312564** to **3127564** (Figure 5, right). We apply Algorithm 2 to both **312564** and **3127564**, and obtain the corresponding FPs to **312564** and **3127564**, respectively. Similarly, for Figure 6 (right), Algorithm 2 is applied to all possible BPs independently.



@R2 - "***the proposed BBP could be compared with methods with sloped cuts to see which type of flexibility is more powerful in real-world relational modeling***": We would like to consider a variant of sloped (oblique) cuts so that the proposed BBP can be extended to a model with the sloped cuts, analogous to the relationship between the binary space partitioning-tree process [19, 25] and the Mondrian process [50]. We agree that recent developments of BNP models with sloped cuts are a very interesting and promising direction of research, and expect that our BPP could also contribute to this issue. We hope that such extension of the BBP could cover the partitions which cannot be easily expressed by the previous models, e.g., pin-wheel structure with sloped cuts shown in the right figure.



@R3 - **Difference between our model (BBP) and the rectangular tiling process (RTP)**: The BPP grows along with the number of blocks, while the conventional RTP grows along with the size of virtual grids (right figure). The RTP constructs a probabilistic generative model that directly generates rectangular partitioning (RP) of grids (i.e., a matrix) with infinite size. As a result, the RTP has too complicated procedures for the model construction, and is not well-suited for Bayesian inference. Specifically, it is very difficult to establish the *effective* Metropolis-Hastings (MH) algorithm for the update steps of the current RP sample drawn from the RTP, while the BPP (and the BBP) can naturally lead to such steps. Consequently, the RTP tends to have low acceptance ratio for such MH steps, which is the most important difference and the practical advantage of the BPP over the RTP.

@R3 - **For experiments,** "***Does each model have a limit on the number of blocks in the partition, or do they run until each partition has only a single label in the training set?***": No, each model is allowed to have unlimited number of blocks without finite truncations. Since Bayesian nonparametric models practically behave like *parsimonious* models, infinite number of meaningless blocks (containing no observation data) are ignored, and *active* blocks must be always finite for finite observation data. For our MCMC inference, the number of active blocks is allowed to be variable order.

@R3 - **For experiments,** "***Can we know more about the properties of the datasets, for example sample sizes?***": As in Lines 230–234, for each real-world data, we employed the *active* 500-by-500 binary matrix extracted from the original very large and *sparse* matrix. For example, **Wiki** data (Wikipedia vote network) [1] consists of 7115 nodes and 103689 edges with diameter 7. We will add this kind of detailed explanations for all data.

@R4 - "***While as a discrete time Markov chain, the BPP is not discussed with its time complexity or other quantitative analysis of computation expense***": **From a viewpoint of generative models** - It takes $\mathcal{O}(K)$ for the BPP (BBP) to draw a FP (RP) with $(K+1)$ blocks given a FP (RP) sample with K blocks. As a result, the evolution of the BPP works with reasonable computational cost, even when significantly many blocks are required. We also emphasize that the conventional Mondrian process (MP) [50] (which can be also expressed as a Markov process) has the same computational order for its evolution. **From a viewpoint of Bayesian inference** - Given an M-by-N input observation matrix, the MCMC calculations plainly involves (a) $\mathcal{O}(K)$: random growth or regression of the Markov process (i.e., updates of the BBP), (b) $\mathcal{O}(M+N)$: random locations of rows and columns of the input matrix, and (c) $\mathcal{O}(KMN)$: data assignments of MN elements to *active* K blocks. (Note: K is allowed to be variable order on each MCMC iteration.) This computational order is same as the MP, which we can also see in the experimental results (Figure 8, top).